

QUANTA

www.quanta.org.uk

CONTENTS

CHAIRMAN'S REPORT.....	John Mason.....	3
COPY DATE for next issue.....		3
EDITOR'S APOLOGY.....	John Gilpin.....	4
QUANTA COMMITTEE PRECIS.....	Sarah Gilpin.....	4
QL & THE GENEALOGIST (3).....	Sarah Gilpin.....	5
MASTERMIND.....	Steve Poole.....	7
JOCHEN MERZ SOFTWARE.....	ADVERT.....	8
PROCedures & FuNctions.....	Steve Poole.....	10
CONFIG BLOCKS IN TURBO (2).....	George Gwilt.....	12
QL TODAY.....	ADVERT.....	14
RWAP QL SOFTWARE.....	ADVERT.....	20
USING SSSS.....	George Gwilt.....	22
QUANTA POSTER.....	ADVERT.....	23
HOVE QUANTA AGM & WORKSHOP.....		26
CLUE to CHRISTMAS CHALLENGE... Steve Poole.....		29
MINISQL VERSUS LCD MONITOR... Dilwyn Jones.....		30
ROBOTICS on the QL (2).....	David Buckley.....	32
QBRANCH.....	ADVERT.....	39
T F SERVICES.....	ADVERT.....	40

Volume 23 Issue 6
Dec 2006/Jan 2007

Information on the Association

Membership of QUANTA, the independent QL user group, is by annual subscription. The Membership Secretary can supply full details. Copies of the association's constitution & annual accounts are available from the Secretary. There is an extensive software library. There are active national subgroups. Details are given in the Member's Guide and in this magazine.

QUANTA Committee - 2006/2007

Chairman John Mason 73 Chewton Common Road, Highcliffe, Christchurch, BH23 5LZ

Email: chairman@quanta.org.uk Tel: 01425 275894

Secretary Sarah Gilpin 181 Urmston Lane, Stretford, Manchester. M32 9EH

Email: secretary@quanta.org.uk Tel: 0161 865 2872

Treasurer ,
Secretary
& Missed Issues John Gilpin 181 Urmston Lane, Stretford, Manchester. M32 9EH Membership

Email: treasurer@quanta.org.uk Tel: 0161 865 2872
Email: membership@quanta.org.uk

Magazine Editor

Email: editor@quanta.org.uk

Librarian Roy Brereton 94 Teignmouth Road, Clevedon, N. Somerset. BS21 6DR

Email: librarian@quanta.org.uk Tel: 01275 871 917

Helpline

Email: helpline@quanta.org.uk

Back Issues Requests Basil Lee 31 Fairdale Gardens, London. SW15 6JW

Email: backissues@quanta.org.uk Tel: 0208 789 1976

QUANTA reserves the right to publish, reject or edit any material submitted. The opinions expressed are solely those of the contributors. Member's private small ads - free (up to 50 words). Non-members private small ads (and members after 50 words) 50p per 10 words. Block ads; Single issue Full page £10, Half page £5, Whole Year Full page £50, Half page £25. Terms strictly cash with order. Receipt if requested. All copyrights and trademarks are hereby acknowledged

Sussex QL User Group was the first sub group to offer to hold the 2007 AGM so the Annual General Meeting will be held at Portslade Town Hall on Sunday 15 April 2007. I look forward to seeing all of you there. Our thanks also go to the West Midlands QL User Group, who offered to host the AGM, but on the last day. We have suggested that they should apply earlier next year.

During 2006 we have carried out extensive trials with an electronic version of the Magazine. These trials have now come to a successful end. Committee have decided to offer Members the option of receiving their magazines electronically as an email attachment in place of the present printed version by post from Volume 24, Issue 1 – Feb/Mar 2007.

The minimum requirement to receive the magazine electronically is a computer with the ability to receive email and a 56K modem.

To subscribe send your email address with the subject "Electronic Magazine 2007" to Sarah Gilpin at secretary@quanta.org.uk as soon as possible please.

Each member who receives their magazine electronically will reduce the magazine postage bill by £1.38 per year for a UK address, more for an overseas member, and this will help to contain the operating costs of Quanta.

**COPY DATE for FEBRUARY/MARCH 2007 ISSUE is:
5TH FEBRUARY 2007**

Please ensure that your copy reaches the editor on or before that date.

Copy on Paper or Magnetic Media or by Email.

All Contributions are Welcome.

Copy on Microdrive or Floppy Disk or CD to be accompanied by hard copy printout.

Copy by Email to - editor@quanta.org.uk

I feel that an apology is needed for the lateness of this issue of QUANTA Magazine. Due to commitments to my family and to my work for the Scout Association, not to mention my work as QUANTA Treasurer, my "spare" time during the last quarter of 2006 has been extremely limited and one of the results of this is the late publication of this issue. My sincere apologies are sent to all our members and traders and I trust that you have not been unduly inconvenienced.

The working committee of QUANTA is currently reduced to three members and with all the will in the world, it is not possible to provide a consistent and efficient service to our members and ideally, a further three members would be welcomed in order to divide the working load between us. By volunteering to take on extra duties for QUANTA, the present committee members stress that the extra work will have to take a lower priority to current commitments and apologises in advance for any delays which are caused by circumstances beyond their control.

I understand from our Secretary that nomination forms sent out with the Oct/Nov 2006 Issue have resulted in only one nomination to stand for election to the committee and that from a non-member who joined QUANTA in order to stand. Surely there must be others who can spare a couple of hours or so per week to share our burden. Although the deadline for nominations has passed, we have the facility to co-opt members onto the committee and those interested would be most welcome. If you know of a member (or yourself) who would be interested in learning more about the work of the committee, please contact our secretary, Sarah Gilpin, or fill in and return a nomination form.

QUANTA COMMITTEE PRECIS

Sarah Gilpin

A committee meeting was held at the Byfleet workshop. The venue and date for the AGM 2007 were discussed. In the absence of any other offers, the AGM will be held at Portslade Town Hall, Sussex on 15th April 2007. A dinner will be organised for Saturday 14th April 2007 if there is enough members interested.

Progress on the electronic version of the magazine was also discussed together with the improvement in format and content since the AGM 2006. It was felt that the trader's opinions of the suggested change in

Research

The increasing interest in family history and the growth in home computers have driven the transfer of old documents into computer format and therefore more information is easily available to the researcher. No longer does all research have to be undertaken in library or cathedral archive offices, nor does the family historian have to be able to read old Secretary handwriting. Parish Registers, Census records, rates records and other civil and ecclesiastical records are being transcribed by students and researchers.

One of the oldest and largest data bases generally available through the Internet is the IGI files generated and held by the Mormon Family History Centres. This information is also available on microfiche at local history centres throughout the UK. The downside to the IGI files is that death is not recorded in the way that baptisms and marriages are. The earliest IGI information was gleaned by word of mouth from the new members of the Mormon Church, and this has led to many variations of spellings and other inaccuracies. The good researcher will check all the data and verify the same from alternative or the original sources.

Information held in the registers of births, deaths and marriages has always been available from Somerset House (more recently St Catherine's House) provided you knew the name and probable year of birth and duplicate certificates are available at a price. The index for this source is now available in many libraries and provides a very useful tool, but although the English/Welsh records are available, in Scotland, the reverse is not true. The limitation is that Civil Registration did not start until 1837 and was not compulsory until 1856. More information was added to the records towards the latter part of the 19th Century, such as the spouse's surname for a marriage, and the mother's maiden name for a birth and the age at death.

Other important sources are the National Census, Rates Books, Parish registers, and records from the Armed forces. One difficulty with many of those records is that they are only available in the locality of origin. A far greater difficulty arises from inaccuracies in the information given to the census enumerator and the vagaries of name spelling by the enumerator due to local accents and dialects. The decision has to be made as to how much research can you do yourself and how much are you willing to pay others to do.

Quanta Magazine

Military records are available in some libraries, and on the Internet. So are lists of emigrants to the USA and lists of criminals to Australia. What is not always available are the names of those who died during the voyage.

For the absolute beginner the place to start is the civil registers and family members. Start with recording the names of your parents and grandparents and their siblings. Include, where known, their dates of birth, marriage and death with places. Armed with this information start to use the Civil registers. All registrations should be made within six weeks of the event; the registrations are listed in quarters Jan – March, April – June etc. Therefore a child born around Christmas may not be registered until January and the record will therefore be in a different year and quarter than expected. The entry in the civil register gives Surname, Forenames (or Initials), Registration District, Volume Number and Registration Number – from which any individual record may be obtained from St Catherine's House.

Another excellent source is parish registers, but where these have been transcribed by such organisations as the Mormons errors occur. The most common error comes from the parish register dates as the church year started April 1st and ran to March 31st. Therefore in the early registers the months Jan to March are often recorded by researchers in the wrong year (One before the actual year). There are two other problems arising from the registers themselves, the first is that until the late 17th early 18th century only the father's name was recorded, and the second one is the common practice of naming younger children after deceased older children. There was also a convention used in the naming of children in many parts of England and Scotland (eldest son named after paternal grandfather, 2nd son after maternal grandfather and 3rd son after father, similarly for girls), so that a number of children in each generation were frequently called by the same name. Although these children may well have been called within the family by shortening or nicknames, the registers only give the original name. Example – Richard Good has 4 sons, say William, Walter, Richard and John; if each of these sons married and had at least one son there would be 4 Richard Goods, their children would show in the baptismal records as name, son of Richard Good, (which Richard, when they all live in the same parish?)

Once the 1881 census became available on disc, whether through the local history centre, at the library or bought for use at home, it became easier to find locations of your ancestors as both place of birth and place of census registration are given. It was at this stage that we realised that we

Quanta Magazine

were gathering information from a number of different sources, but had no cross reference to the source of any single piece of information.

Dilwyn Jones has produced an excellent list of sources ¹ for family history research. There are also many research organisations on the Internet, such as Genes Reunited, Roots and many more. There are also sites available for a single surname or for a place name, e.g. Stonehaven. Some sites are free for browsing, but charge for printed data while others have a registration fee before information can be accessed. Many towns, counties and regions have their own historical society which researches and records local information, but many have a joining fee. Many of these have publications and CDs at a reasonable price for their area.

One such organisation told all new members that for earlier family accuracy there should be a six point reference for any individual. The parents and both sets of grandparents of an individual make up the six point reference and is particularly helpful for the 17th and 18th centuries, where the naming of children was so often duplicated.

Armed with this research information, and source material the family historian is ready to record and cross reference the same.

The final article in this series will describe the methods that we initially used for recording our information some 10 years ago and the refinements we have made as we became more proficient and made use of the Internet sources. We have spent time in Somerset, Gloucestershire, Berkshire, Oxfordshire and Aberdeenshire conducting research; some to verify existing information and some to find out new information. All these sources need to be recorded on the family database.

Notes: ¹ Some years ago, Dilwyn Jones provided me with a floppy disc containing a number of sources which I found very useful. The same disc contained a Viewer programme which enabled me to select the individual

MASTERMIND

Steve Poole

One of the all-time favourite computer-games was the famous 'Mastermind'. Perhaps one of the most surprising aspects of the game was that it needed such little code to simulate. Most of the code in the listing is as usual for initialisation and input error-trapping. (To avoid copyright restrictions the game does not conform strictly to the commercial version).

JOCHEN MERZ SOFTWARE

Kaiser-Wilh.-Str. 302 D-47169 Duisburg

Tel. 0203 502011 Fax 0203 502012

<http://smsq.j-m-s.com> smsq@j-m-s.com

SMSQ/E

QPC2 Version 3 with SMSQ/E ... now only **EUR 59,90**

QPC2 Version 3 upgrade from Version 2 ... only **EUR 19,90**

QPC2 Version 3 upgrade from Version 1 ... only **EUR 39,90**

QPCPrint ... still only **EUR 39,90**

Please note that QPCPrint will be shipped on CD only.

QPC2 will also be shipped on CD by default now, as more and more systems (especially notebooks) do not have floppy disks drives built in anymore. If you prefer QPC2 on HD floppy disks, please state with your order.

Free updates are available on Marcel's Website www.KILGUS.net

If you prefer updates on CD or floppy disk, send in your master medium together with 4 international reply coupons to cover return postage, medium and packaging.

SMSQ/E for ATARI or GoldCard or SuperGoldCard or QXL ...

now only EUR 39,90

Colour drivers now included as standard for (Super)GoldCard versions

For shipping inside Europe, EUR 3,- will be added for postage and packaging.

For shipping outside Europe, EUR 6,- will be added for p & p.

If you wish to place your order via internet, please do not send your credit card details in an email!

Please use the secure contact form on the SMSQ homepage: SMSQ.J-M-S.COM

We accept VISA, MasterCard and Diners Club!

Quanta Magazine

Typically, just one REPEAT loop containing two nested FOR-loops can do just about anything...

To play, just RUN and follow the instructions. Two pins and two colours are easy, but nine pins and nine colours are diabolical! For example, nine numbers have factorial 9! combinations, being 362880 possible answers! (The colours are simply represented by numbers to simplify the coding).

Remember, to play you must INPUT your try as consecutive figures in the range 1 to 9 (or whatever the number of figures you have selected). The number of pins is initially indicated by the number of full-stops...

If you don't know how to play, just ask an older friend who will certainly be able to explain!

My thanks go to Bruno Coativy who beta-tested this program and found a nasty bug, which I have eliminated by using mask-strings to avoid counting pins twice. This bug occurred because I had forgotten to test the code with fixed value-limits...Aieee!...haste and over-confidence...

Happy mind-mastering!

```
100 ::
110 REMark Mastermind_bas. by S.Poole, v20dec2006
120 REMark Beta-Test & Optimisation by B.Coativy
130 REMark for Quanta
140 :
150 OPEN#1,con_8: WINDOW 256,206,256,0
160 PAPER 2: INK 7: CLS: ct=1
170 AT 0,1: CLS 3: INPUT'Difficulty? (2 to 9)!'!d$
180 IF d$ INSTR 'Qq': STOP
190 IF d$ INSTR '23456789': n=d$: ELSE GO TO 170
200 r$='': FOR f=1 TO n: r$=r$&RND(1 TO n)
210 AT 0,1: CLS 3: PRINT 'no.','try','good','mis-placed'
220 a$='123456789': a$=a$(1 TO n)
230 AT 1,7: PRINT FILL$('.',n)
240 :
250 REPEAT loop
260 PRINT TO 1;ct; TO 7;: INPUT;t$; : good=0: miss=0
270 IF t$ INSTR 'Qq': STOP
```

Quanta Magazine

```
280 IF LEN(t$)<>n: PRINT! 'Oops!': NEXT loop
290 :
300 FOR i=1 TO n
310     IF NOT(t$(i) INSTR a$): PRINT! 'Oops!': NEXT loop
320 END FOR i
330 :
340 IF t$=r$: PRINT! 'Bravo!': EXIT loop
350 b$=r$
360 REMark 1st pass: see if correct
370 FOR f=1 TO n
380     IF r$(f)=t$(f) THEN
390         good=good+1: b$(f)="@": t$(f)="*"
400     END IF
410 END FOR f
420 :
430 REMark 2nd pass: see if mis-placed
440 FOR f=1 TO n
450     p=t$(f) INSTR b$
460     IF p: miss=miss+1: b$(p)="@"
470 END FOR f
480 :
490 PRINT TO 18!good; TO 25!miss: ct=ct+1
500 END REPEAT loop: i$=INKEY$(#1,500): RUN
510 ::
```

End of listing

PROCedures & FuNctions

Steve Poole

Have you ever wondered why the FuNctions CHR\$, DATE\$, DAY\$, FILL\$, INKEY\$, SDATE\$ or VER\$ end with a dollar-sign? Or why (x DIV y), (x MOD y) or (INT(x)) aren't written (x DIV% y), (x MOD% y), or (INT%(x)). Don't look for the answer in the Beginner's Guide or Jan Jones's SuperBASIC Handbook : no mention is made of them. To find the answer, I have gleaned snippets of information from various listings, and have managed to put the various pieces together, so here are my findings:

There are three sorts of SuperBASIC FuNctions : float, integer and string. Usually most listings only use one type : float. We shall see why later. But first a description of SuperBASIC routines : float, Integer & String-type routines are identified by their names. Float has no suffix, Integer has a '%' suffix and string has a '\$' suffix. Float routines must RETURN a floating-point expression, Integer routines must RETURN an Integer% expression, and String routines must RETURN a string\$ expression. (But this is perhaps not

Quanta Magazine

the case under SMSQ/E). That seems simple enough and is illustrated by the routines on lines 150 and 190.

So why are these constructions not described for us Joe Bloggs programmers? The TURBO Manual uses the three different routine types, and even gives clues such as the BASIC_TYPE% keyword, which describes what it does but not how it does it! (The routines appear to be written in machine code...)

The answer seems to be that the writers of the SuperBASIC Language did not have enough time to describe things fully by the time that the QL was launched. And the QL by default supposedly converts everything to float-type anyway! Now for the results I found:

SuperBASIC is a very flexible language, but it did not make full use of such advantages as integer FOR-loops% or String SElect\$. We had to wait for TURBO to get these. When Basic routines were defined, flexibility went haywire : look at the routine on Line 280 : what a confusing mess! To make any sense of it, look at line 130 : the actual calling parameters of the FLOAT routine are all of string type. And here is the rub : the type of the routine's Formal parameters is determined by the type of the actual calling parameters, not by any suffixes they might have. This means that the variables on line 290 are all of string type, even though they appear to be float, integer and string! I said it was a mess....so just read that last paragraph again!

The lesson is, of course, to give formal parameters the type you expect them to receive! Don't use all the flexibility (unless you are prototyping), and use suffixes to make the function of Procedures, functions, parameters and variables clear. Coercion on the QL is very much complicated by all the many permutations that calls and assignments can adopt.

Maybe I will write another article on the thorny problem of LOCAL parameters, (not LOCAL variables), and the LOCAL nature of names, (not values)! And perhaps another on the curious use of brackets under basic...and unexpected calling orders...The list of SuperBASIC Routine Quirks is a long one.

Best Wishes,

Steve Poole.

I already have an article on brackets (parentheses) from Steve and this will appear in a later issue [Ed]

Quanta Magazine

```
100 ::
110 REMark PROC_FN_bas
120 CLEAR: OPEN#1,con_32: WINDOW 256,206,256,0: CLS
130 PRINT integer%(PI),string$(1),float('abc','d','e','f')
140 :
150 DEFine FuNction integer%(float)
160 int%=INT(float): RETurn int%
170 END DEFine
180 :
190 DEFine FuNction string$(float)
200 s=float
210 SElect ON s
220 =0: RETurn '0'
230 =1: RETurn '+1'
240 =REMAINDER : RETurn '-1'
250 END SElect
260 END DEFine
270 :
280 DEFine FuNction float(var,num,num%,num$)
290 num=var(1): num%=var(2): num$=var(3)
300 Lg=LEN(num&num%&num$) : RETurn Lg
310 END DEFine
320 ::
```

End of Listing.

CONFIG BLOCKS IN TURBO(2)

George Gwilt

The following article is reproduced by kind permission of the author and The Scottish QL Users Group (SQLUG) in whose July & August Newsletter it was first published. It is the concluding part of the article printed in QUANTA Magazine Volume 23 Issue 4 page 22 and was held over from Issue 5 due to shortage of space. [Ed.]

New code added for Codegen_task v4.07

This allows an "extension" with first parameter 1 to be used to insert an _INS file. First the markers "\$£#" (for the INS position) and "((TZ>>" are found. If these are there, the INS file name from the correct "extension" is loaded to RAM. If this is a correct file the contents are placed at the INS position.

* D5 is a state marker. We look for A "\$£#" and B "((TZ>>".

* State -1 = only B found

Quanta Magazine

* 0 = neither found
* 1 = only A found
* D3 = -1
* D4 = "TZ>>"
* D6 = "#*\$£"
* D7 = "("

```
        movea.l  obj_chan,a0    ID of the file of object code
        moveq   #0,d1          -> start of file
        moveq   #fs_posab,d0
        trap    #3
        moveq   #0,d5          mark state 0
        move.l  #"TZ>>",d4
        move.l  #"#$£",d6
        move.w  #"(",d7
        moveq   #-1,d3
        subq.l  #4,sp          make a space for a long word
lk      bsr     get_wd         set a word to stack
        tst.b   d5
        beq    lk_1          state 0
        bmi    lk_3          state -1
;
;Operations for state 1
;Now looking for "B"
        cmp.w   (sp),d7       Could it be "B"? . .
        bne    lk          . . no, try again
        bsr    get_lwd       set a long word to stack
        cmp.l  (sp),d4       Is it really "B"? . .
        bne    lk          . . no, try again
        bsr    do_pos        find B's current position . .
        move.l  d1,4(a1)     . . and store it
        bra    lk_end        Finished!
; Operations for state 0
; Looking for both "A" and "B"
lk_1    cmp.w   (sp),d6       could it be "A"? . .
        beq    lk_1a        . . yes
        cmp.w  2(sp),d7     could it be "B"? . .
        beq    lk_1b        . . yes
        bra    lk          not found so try again
; First word of A just found
lk_1a   bsr     get_wd
        swap   d6
        cmp.w  (sp),d6
        beq    lk_1c        "A" found
```

Quanta Magazine



QL Today

The magazine for all QDOS SMSQ users
published 5 times a year.

Volume 11 issue 2 is Now Out !

We apologise for the fact that it was a little late this time and hope to make
up for it by getting volume 3 out much more promptly.

IF YOU HAVE NOT YET RE-SUBSCRIBED WE DO HAVE ALL BACK ISSUES
AVAILABLE SO YOU NEED NOT MISS A THING.

If you do not already get QL Today why not get a sample issue and try it
now ?

You can subscribe by contacting :

Jochen Merz Software
Kaiser-Wilhelm Str. 302, 47169 Duisburg, Germany
Tel +49 (0) 203 502011 Fax +49 (0) 203 502012
email : jmerz@j-m-s.com

Q Branch
20 Locks Hill Portslade, Sussex, BN41 2LB. UK
Tel +44 (0) 1273 386030 (eves) Fax +44 (0) 1273 430501
email : sales@qbranch.demon.co.uk

web : www.qtoday.com

Quanta Magazine

```
        swap    d6          restore D6 for another day
        bra     lk
; A now found
lk_1c   moveq   #1,d5       set to state 1
        bsr    do_pos      find A's position . .
        move.l d1,(a1)     . . and store it
        bra     lk         go back to find "B"
; Could be B
lk_1b   bsr    get_lwd     get the next long word
        cmp.l  (sp),d4     is it the rest of "B"? . .
        bne   lk         . . no. Try again
        moveq  #-1,d5      set to state -1
        bsr    do_pos      find B's position . .
        move.l d1,4(a1)    . . and store it
        bra     lk         Go back to search for "A"
; Operations for state -1
; Now searching for "A"
lk_3    cmp.w  (sp),d6     Could it be A? . .
        bne   lk         . . no, try again
        swap  d6
        bsr   get_wd      Get next word
        cmp.w (sp),d6     Does it complete A? . .
        beq  lk_3a       . . Yes
        swap  d6         Restore D6 and . .
        bra   lk         . . try again
; Subroutines and error routines
; DO_POS1 advances D1.L - 6 bytes and sets A1 -> zeroes
do_pos1 subq.l  #6,d1
        bra   do_pos2
; DO_POS sets the current position to D1.L and sets A1 -> zeroes
do_pos  moveq  #0,d1
do_pos2 moveq  #fs_posre,d0
        trap  #3
        lea  zeroes,a1
        rts
; GET_SIX, GET_WD and GET_LWD set 6, 2 and 4 bytes from the file to
the stack
get_six moveq  #6,d2
        bra  get_wd1
get_wd  moveq  #2,d2
        bra  get_wd1
get_lwd moveq  #4,d2
get_wd1 lea   4(sp),a1     Skip the return address
```

Quanta Magazine

```
get_wd4    moveq    #io_fstrg,d0
           trap     #3
           tst.l    d0
           bne     get_wd2
           rts

; Error entries
d_ins_error addq.l    #6,sp          Tidy stack
d_ins_error1
           lea     ins_mess,a0      Send INS error . .
           bsr     error
           bra     fatal2          . . message

; Probably EOF (shouldn't happen)
get_wd2    addq.l    #4,sp
           subq.w   #6,d2          was it later? . .
           beq     d_ins_error1    ----> . . yes
           bra     lk_end          defer error message

; "A" found
lk_3a     bsr     do_pos
           move.l   d1,(a1)

;
lk_end    addq.l    #4,sp          Tidy stack
* now the position of the INS marker is at zeroes and of the extension
* marker at zeroes+4.
* We search the extension area for the name of the INS file, check it
* and load it into the INS place.
*
* In each case the file position is set just after the marker.
* To find the correct extension we read in 6 bytes. The first four
* give the relative position of the next file, or -1 to signal the end.
* The next word is 1 for the required extension.
* When the extension is found, the name of the INS file is two bytes from
* the end of the "1".
           subq.l    #6,sp          space for info from file
           lea     zeroes,a1
           tst.l    (a1)
           beq     d_ins_error    ----> no INS to do
           move.l   4(a1),d1      pointer to EXTENSION info
           beq     d_ins_error    ---->
           moveq   #fs_posab,d0
           trap     #3
d_ins1    bsr     get_six          get 6 bytes
           move.l   (sp),d1       to next file
           bled_ins_error    ---->
```


Quanta Magazine

```
    cmpi.w    #1,4(sp)    _INS name? . .
    beq      d_ins_found  . . yes
    bsr      do_pos1      go forward D1.L - 6 bytes
    bra      d_ins1

d_ins_found bsr      get_lwd      get length of INS filename .
            move.w    2(sp),d2    . to D2.W
            addq.l    #6,sp      reset stack
            lea      zeroes+92,a1
            move.w    d2,(a1)+    set length for OPEN
            bsr      get_wd4     get filename to zeroes+90
*
* Now compare names (without regard to case)
* We put the name in INS place to zeroes + 82.
* We put (truncated to 10 bytes) the name from the filename
* to zeroes + 142, with the count in zeroes + 140.
* This count is also put in zeroes + 80.
*
* The comparison is by UT_CSTR
*
            move.l    zeroes,d1    position of name in INS . .
            addq.l    #2,d1        . . set
            moveq     #fs_posab,d0
            trap      #3
            moveq     #10,d2       10 bytes needed
            lea      zeroes+82,a1
            bsr      get_wd4       get name from INS place
            moveq     #"_",d4
            lea      zeroes+90,a2
            adda.w    zeroes+92,a2  -> end - 4
            move.l    a2,d5        keep place in D5.L
            move.l    d5,d2
            lea      zeroes+95,a0
            sub.l     a0,d2
d_ins2      cmp.b     -(a2),d4
            dbeq     d2,d_ins2     look for "_"
            bne      d_ins_error1  ---->
            addq.l    #1,a2
            sub.l     a2,d5        length
            cmpi.l    #10,d5
            bled_ins5
            moveq     #10,d5       restrict length to 10
d_ins5      lea      zeroes+140,a0 place for restricted name
```

Quanta Magazine

```
move.w    d5,(a0)+      first the length . .
lea       zeroes+80,a1
move.w    d5,(a1) . . also to the INS name
bra       d_ins6
d_ins3    move.b      (a2)+,(a0)+
d_ins6    dbf         d5,d_ins3
lea       zeroes+140,a0  reset A0
suba.l    a6,a0
suba.l    a6,a1
moveq     #1,d0          comparison (any case)
movea.w   ut_cstr,a2
jsr       (a2)
tst.l     d0
bne       d_ins_error1  ---->

d_ins4    lea         zeroes+92,a0  point to filename
moveq     #1,d3          open_in
moveq     #io_open,d0
moveq     #-1,d1        myself
trap      #2
lea       zeroes+16,a1  for header
move.l    a0,-4(a1)     keep ID of INS file
moveq     #-1,d3        timeout
moveq     #64,d2        size of buffer
moveq     #fs_headr,d0
trap      #3
move.l    zeroes+16,d1  length of file
moveq     #-1,d2        myself
moveq     #mt_alchp,d0
trap      #1
tst.l     d0
bne       d_ins_error1  ---->
lea       zeroes+8,a1
move.l    a0,(a1)       keep ALCHP'd address
movea.l   a0,a1
move.l    zeroes+16,d2  length of file
moveq     #-1,d3        timeout
movea.l   zeroes+12,a0  ID
moveq     #fs_load,d0
```

CONFIG BLOCKS IN TURBO (CONCLUDED)

ERRATA In the previous articles £ should be £

Quanta Magazine

```
trap      #3
moveq    #io_close,d0    å
trap      #2              close INS file
movea.l  obj_chan,a0
moveq    #-10,d1         to wind back 10 bytes
moveq    #fs_posre,d0
trap      #3
move.l   zeroes+16,d2
moveq    #-1,d3
movea.l  zeroes+8,a1
moveq    #io_sstrg,d0
trap      #3
movea.l  zeroes+8,a0
moveq    #mt_rechp,d0
trap      #1
```

```
*****
*      End of code for _INS files      *
*****
```

Operating System Calls

Trap #1 calls

mt_alchp Reserves space and returns its address

Entry	Exit
D1.L is the size needed	A0 is the address
D2.L is set to -1 for "this job"	

mt_rechp Returns space to the heap

Entry	Exit
A0 is the address	

Trap #2 calls

io_open Opens a file and returns its ID

Entry	Exit
D1.L is set to -1 for "this job"	A0 contains the channel ID
D3.L is set to 1 for "open_in"	
A0 points to the file name	

io_close Closes a channel

RWAP Services NEWS!

We have moved back to Walsall for the time being - see our current address at the bottom of the page.

Better news is that we now have an extended range of second hand items for the Sinclair QL and ZX Spectrum, and have one remaining Epson 850 Inkjet printer available. Z88 stocks are however now getting quite low so if there is anything you need, please browse our website:

<http://www.rwapsoftware.co.uk>

We are also looking to produce some new hard disk interfaces for the ZX Spectrum and have a few little projects on the drawing board.

Our websites:

<http://www.rwapservices.co.uk> (General site)
<http://www.rwapsoftware.co.uk> (Sinclair computer second hand and new items)
<http://www.rwapadventures.com> (Adventure Programs)
<http://www.internetbusinessangels.com> (Guidance on setting up online businesses)

New Products!

QWORD 1.0

NOW WITH DIGITAL
SOUND ON QPC2!

The wait is now over! Q-Word version 1 is finally available!

Platforms:

QPC/QXL, Q40/Q60, Aurora (with SGC)

Prices:

All versions without P-Word £20.00
 All versions with P-Word £30.00

Notes:

Q-Word DOES NOT require SMSQ/E with GD2 support -OR- SMSQ/E at all on the Aurora or Qx0 machines. It works on the highest colour depth everywhere regardless of Operating System.

The Aurora version is available on either HD or ED disk. For the latter add £1.00 to the price. ED version is uncompressed and can be run directly from the floppy. All other Floppy versions are compressed. QPC/QXL version comes on CD. Non CD versions DO NOW support digital sound on QPC2

Quantum Leap ED Drives

The bad news is that our stock of ED Disk Drives has now been depleted and there is no sign of any more being available in the short term.

We do however have a range of brand new DD and ED 3.5" Diskettes for sale at low prices:

10 x 3.5" DSDD Disks £7.50
 10 x 3.5" ED Diskettes £10
 10 x 3" Disks £9

Prices do not include post and packing - please ask us for details.

We do have a range of second hand disk interfaces and drives (one or two ED Drives) for use with the Sinclair QL, so if you need anything, please let us know.



for Windows

For QLers that run Windows or with incompatible hardware for Talent Games, we now have re-released these adventures so that they can run on your Windows-equipped PC. No Emulator, floppies, microdrive backups etc. required, just a one-click install! Of course the full QL line is still available! (See side column)

Talent Games for Windows ea. £ 10.00
 (Each Game includes a runtime installation of QLAY-2 by Jimmy Montesinos)

RWAP Services

26 Oak Road, Shelfield, Walsall, West Midlands W54 1RQ TEL: +44 1922 691607 From the UK Dial: 01922 691607
 Website: <http://www.rwapsoftware.co.uk>
 Email: sales@rwapsoftware.co.uk

We Accept Payment using:



(Cheques in £ sterling made payable to R. Mellor)

Old Favourites!

Utilities	
SBASIC / SuperBASIC Reference Manual on CD	£ 20.00
Sidewriter v1.08	£ 10.00
Landscape Printing (EPSON printers)	
ImageD v1.03	£ 10.00
3D object generator	
Q-Help v1.06	£ 10.00
Superbasic On-Screen help system	
Q-Index v1.05	£ 5.00
Keyword-to-topic finder	
ProForma ESC/P2 Drivers v1.04 for ProWeSe Printer Driver	£ 8.00

Applications

Flashback SE v2.03 (upgrade only)	£ 5.00
Database	
QL Cash Trader v3.7	£ 5.00
Accounting/Finance	
QL Payroll v3.5	£ 5.00
Accounting/Finance	
QL Genealogist v3.26	£ 20.00
Genealogy	
Genealogy for Windows	£ 50.00
QL Genealogist to Windows version upgrade	£ 25.00
QL Cosmos v2.04	£ 5.00
Planetarium	
Q-Route v2.00	£ 25.00
Route Finding	
Upgrade from v1.xx	£ 5.00
Britain map v1.11	£ 2.00
BIG Britain map (needs 2Mb) v2.03	£ 5.00
Various Britain Area maps (ask for details)	ea. £ 2.00
Ireland map v1.00	£ 5.00
Belgium map v1.01	£ 2.00
Catalonia map v1.03	£ 2.00
P-Word UK English Dictionary (500.000 words) Dictionary	£ 15.00

Leisure

Return to Eden v3.08	£ 10.00
Adventure	
Nemesis MkII v2.03	£ 8.00
Adventure	
The Prawn v2.01	£ 8.00
Adventure	
Horrorday v3.1	£ 8.00
Adventure	
West v2.00	£ 5.00
Adventure	
The Lost Kingdom of Zkul v2.01	£ 5.00
Adventure	
All 6 games above	£ 25.00
D-Day MkII v3.04	£ 10.00
Strategy/War Simulation	
Grey Wolf v1.08	£ 8.00
Graphical Submarine Simulation	
War in the East MkII v1.24 (upgrade only)	£ 5.00
Strategy/War Simulation	
Open Golf v5.20	£ 8.00
Sports Simulation	
QuizMaster II v2.07	£ 5.00
Quiz	
Stone Raider II v2.00	£ 5.00
Arcade Game	
Hoverzone v1.2	£ 5.00
Arcade Game	
Deathstrike v1.5	£ 5.00
Arcade Game	
Flightdeck v1.0	£ 10.00
Flight Simulation	
All 6 games above (Open Golf, QuizMaster II, Stone Raider II, Hoverzone, Deathstrike and Flightdeck)	£ 28.00

Notes on Software requirements

The following programs have a minimum SGC card requirement: P-Word, Qword, Big Britain MAP for Q-Route

Quanta Magazine

Entry
A0 is the channel ID

Exit

Trap #3 calls

In all these calls A0 must contain the channel ID on entry and D3.W contains the timeout. In the above code timeout is always -1 (forever).

io_fstrg Fetches a number of bytes to a buffer from a channel

Entry Exit
D2.W is the number of bytes D1.W is the number
fetched
A1 points to the buffer's start A1 points to the end

io_sstrg Sends a number of bytes from a buffer to a channel

Entry Exit
D2.W is the number of bytes D1.W is the number sent
A1 points to the buffer's start A1 points to the end

fs_posab Sets the position in a file

Entry Exit
D1.L is the requested position D1.L is the new position

fs_posre Sets the relative position in a file

Entry Exit
D1.L is the offset required D1.L is the new position

fs_headr Reads the file header to a buffer

Entry Exit
D2.W is the buffer length D1.W is the number read
A1 points to the buffer's start A1 points to the end

fs_load Loads a file into a buffer

Entry Exit
D2.L length of file
A1 points to the buffer's start A1 points to the end

Vector

ut_cstr Compares two strings

Quanta Magazine

Entry	Exit
D0.B is the comparison type	D0.L = 0 if strings same
(A0,A6.L) -> 1st string	
(A1,A6.L) -> 2nd string	

Comparison types are from 0 to 3. Type 1 compares strings without regard to case.

USING SSSS

George Gwilt

SSSS is the SMSQE sampled sound system provided on the Q40, the Q60 and QPC2. I would guess that not many people have made much, if any, use of this facility. After all, if anyone wants to incorporate a sound in their program they probably use BEEP, which certainly can produce a wide variety of interesting noises and which works on all QL types unlike this curious SSSS.

First of all I should explain what a sampled sound system is. Briefly it is a method of converting sounds between analogue and digital form. Sound picked up by a microphone will appear as a voltage varying with time. This is captured at discrete intervals of time and converted to a number by a piece of hardware called an analogue to digital converter (ADC) and recorded. Later this set of numbers can be reconverted to analogue form by a piece of hardware called a digital to analogue converter (DAC). There are no ADCs on a Q40 or Q60 but there are two DACs, one for the left speaker and one for the right. You might now see that the "sampled" part of SSSS refers to the fact that the sound is sampled at intervals. The frequency of sampling obviously has an effect on the fidelity of the digital representation. Higher frequencies mean better recordings but larger files of numbers. The frequency used by SMSQE is 20k Herz which is considered sufficient for most purposes.

With the Q40/60 come programs, all based on SSSS, by Claus Graf, Simon Goodwin and Jonathan Hudson, which will play sound. What I want to do here is to show how anyone can use this sound system directly without needing, for example, Simon Goodwin's SSSS device driver. I must say that I am doing this partly out of perversity since it is suggested in the one and a half page description of the SSSS that "in principal this [program] will be a sound device". What I will show is rather simpler.

It is clear that SSSS will only play files which already exist; it will not help you to produce these. In later articles I hope to suggest how this may be done. For the moment I am concerned only with the process of using SSSS to play existing files.

QUANTA



Independent QL Users Group

World-wide Membership is by subscription only,

Offering the following benefits:

Bimonthly Newsletter – up to 40 pages

Massive Software Library – All Free!

Free Helpline and Workshops

Regional Sub-Groups. One near you?

Advice on Software and Hardware problems

Subscription just £14 for UK members

Overseas subscription £17

Barclaycard: Visa: Access: MasterCard: Accepted

Now in our Twenty Third Year

Further details from the Membership Secretary

**John Gilpin, 181, Urmston Lane
Stretford, Manchester, M32 9EH (UK).**

Tel. +44 (0) 161 865 2872

Or

Visit the Quanta Web Site

<http://www.quanta.org.uk>

E-mail: membership@quanta.org.uk

Next QUANTA Sponsored Event

Sussex QL User Group

Present:

The HOVE QUANTA AGM & Workshop

At Portslade Town Hall

Victoria Road,

Portslade,

Sussex. BN41 1YF

On

Sunday 15th April 2007 10am till 4pm

Watch later QUANTA and QL Today Magazines for full details.

Quanta Magazine

I must now explain what SSSS allows you to do. SSSS maintains a queue of byte pairs which, when the sound is on, are sent successively to the left and right DACs. Each pair of bytes in the queue is called a sample.

There are five entries into the system to control operations.

Operation	Entry
1. To add one sample to the queue.	JSR \$04(A2)
2. To add multiple samples to the queue.	JSR \$08(A2)
3. To notify SSSS that multiple samples have been added.	JSR \$0C(A2)
4. To stop the sound.	JSR \$10(A2)
5. To enquire how many samples remain in the queue.	JSR \$14(A2)

What I have done is to write a small assembler program which can be CALLED by S*BASIC instructions to use SSSS. There are three entries to the program.

The first performs operations 2 and 3 above. For this the program takes as parameters an address of an area in RAM containing a set of bytes which will be treated as mono sound. That is, each byte will be sent both as left and as right. The second parameter is the number of bytes. SSSS has a \$32000 byte buffer to hold the sound bytes. To load multiple samples operation 2 is called. This sets the start and finish of the available space in this buffer. When the program has sent a batch of bytes to this buffer, it notifies SSSS by operation 3. This tells SSSS to "play" these bytes.

The second entry has no parameters and just stops the sound.

The third entry has no parameters either, but it returns the number of samples remaining in the queue.

Here is the program, followed by some comments:

```
; CALL asad,a,k adds multiple samples
; (a is the address of a k-byte area of sound pairs)
;
; CALL asad+2 stops the sound
; CALL asad+4 sets the number of samples left to asad+6
;
;
; The macro SETVEC is used for START2 and START3
```

```
SETVEC MACRO ENTRY
```


Quanta Magazine

```

MOVEA.L $70,A3           Level 4 auto vector
MOVEA.L -(A3),A2        set the required vector to A2
CMPI.L  #"SSSS",- (A3)  Is it the sound system?
BNE     BAD_EXIT        ---->
TRAP    #0              to supervisor mode
JSR     \1(A2)          do the job
MOVE    #0,SR          back to user mode
ENDM

```

NOLIST

```
IN      WIN1_SYS_EQUBAS_ASM
```

LIST

```

BRA     START1         send samples
BRA     START2         kill sound
BRA     START3         returns number of samples left . .
ANS     DS.L           1 . . in ANS

```

```
START1  MOVEA.L D1,A5           address to A5
```

;

```

MOVEA.L $70,A3
MOVEA.L -(A3),A2
CMPI.L  #"SSSS",- (A3)
BNE     BAD_EXIT        ---->
MOVEQ   #0,D5           mark "not finished yet"
MOVEA.L A2,A4           keep vector
TRAP    #0
JSR     8(A2)           prepare for multiple entry
MOVE.L  D2,D4           set k to D4
MOVEA.L A5,A0           start of space

```

```

BRA     L1
L3     SWAP            D4           switch to low word (65535)
L      MOVE.B         (A0),(A1)+   same byte . .
      MOVE.B         (A0)+,(A1)+   . . to left and right
L1     CMPA.L         A2,A1         Is buffer full? . .
      BGE            L2           . . yes (before time maybe)
L4     DBF            D4,L         count k pairs (low word)
      SWAP            D4           prepare to count high word . .
      DBF            D4,L3        . . count high word
      MOVEQ           #1,D5       mark "finished"

```

```

L2     MOVEA.L         A4,A2
      JSR             $(A2)       tell SSSS that samples were added
      MOVE            #0,SR       back to user mode
      TST.L           D5         finished? . .
      BNE            EXIT1       . . yes
      BSR            SUS         wait 5 ticks . .

```

Quanta Magazine



The Hove Quanta AGM & Workshop

Victoria Road Portslade Sussex, BN41 1 YF

Sunday, 15th April, 2007

10am till 4pm

Welcome to the 2007 Hove Quanta AGM & Workshop.

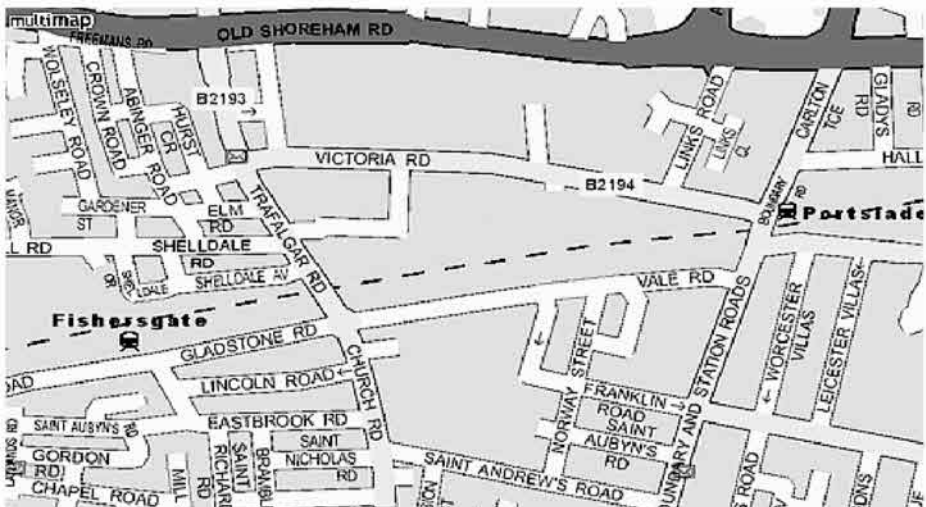
This year we once again hope to have several interesting talks to engage your interest.

Details will be announced closer to the date.

We are arranging a dinner on the Saturday evening for those staying overnight. This will be held at Bom Banes Restaurant. (www.bom-banes.co.uk) This is a most unusual venue and should be a good evening. Please contact sales@qbranch.demon.co.uk to book a place.

All this, the usual good food at the venue and a day by the seaside.

A list of hotels will be posted soon. What more could you ask for?



Quanta Magazine

```
        MOVEA.L  A4,A2          . . replace vector pointer . .
        TRAP     #0             . . to supervisor mode . .
        JSR      8(A2)          . . get new queue pointers . .
        BRA     L4              . . and try to finish this time.
EXIT1   MOVEQ    #0,D0
        EXIT     RTS
BAD_EXIT MOVEQ   #-15,D0
        BRA     EXIT
;
START2  SETVEC   $10
        BRA     EXIT1

START3  SETVEC   $14
        LEA    ANS,A0
        MOVE.L D0,(A0)
        BRA     EXIT1

SREG    REG      D0-1/D3/A0
SUS     MOVEM.L  SREG,-(A7)
        MOVEQ   #-1,D1          this job
        MOVEQ   #5,D3          wait 5 ticks
        SUBA.L  A1,A1          notify no-one
        MOVEQ   #MT_SUSJB,D0
        TRAP    #1             wait
        MOVEM.L (A7)+,SREG
        RTS
```

The file which is included in the program follows:

```
; EQUBAS_ASM on WIN1_SYS_
; 7 AUG 92
;
BV_BFBAS EQU 0
BV_BFP EQU 4
BV_TKBAS EQU 8
BV_TKP EQU $C
BV_NTBAS EQU $18
BV_NTP EQU $1C
BV_NLBAS EQU $20
BV_NLP EQU $24
BV_VVBAS EQU $28
BV_VVP EQU $2C
BV_CHBAS EQU $30
BV_CHP EQU $34
BV_RTBAS EQU $38
```

Quanta Magazine

BV_RTP	EQU	\$3C
BV_LNBAS	EQU	\$40
BV_LNP	EQU	\$44
BV_BTP	EQU	\$48
BV_BTBAS	EQU	\$4C
BV_TGP	EQU	\$50
BV_TGBAS	EQU	\$54
BV_RIP	EQU	\$58
BV_RIBAS	EQU	\$5C
BV_SSP	EQU	\$60
BV_SSBAS	EQU	\$64

Comments on the Program:

1. The sound system subroutines are found by examining an address, a, stored in the level 4 auto interrupt which is at RAM address \$70. The long word at a-8 is "SSSS" and the long word at a-4 is the address used for the entries to SSSS.

The program sets the address a in A3, the address in a-4 to A2 and checks that the contents of a-8 are indeed the bytes "SSSS".

2. All accesses to SSSS must be in supervisor mode, which is achieved by TRAP #0. Return to user mode is brought about by MOVE #0,SR.

3. Since the number of byte pairs to be sent to SSSS may exceed $2^{16} - 1$ (65535) it is necessary to modify the usual method of counting. This is done here by counting the low word in D4. We drop through the DBF when D4.W becomes -1. At this stage we swap the words of D4 and count the high word with a second DBF. If the high word of D4 was not zero we jump to L3 where the words of D4 are again swapped, leaving D4.W equal to 65535.

4. The entry JSR \$08(A2) to SSSS sets the start address of the available buffer area to A1 and the end of this area to A2. We thus have to check after each byte pair has been sent to the buffer whether there is room for more. If there is not room SSSS is notified that multiple bytes have been sent so that it can play these and free the buffer for more bytes later. At this stage we wait five ticks and return to L4 to send a further batch of sound bytes.

5. The program can easily be adapted to play stereo sound by altering the line at L to:

```
L    MOVE.B  (A0)+,(A1)+
```

Quanta Magazine

This means of course that the size of the area of sound pairs must be doubled for the same time length of sound.

6. This program can be tested using any set of bytes. You can even use RAM from 0 upwards! The sound you get will be probably be no more than rasping. It will certainly not be Beethoven's Ninth Symphony.

CLUE TO THE XMAS CHALLENGE

Steve Poole

We know that it's not 28th February yet but just in case you are struggling with the QUANTA Christmas Challenge (Volume 23 Issue 5 Page 9) Steve has sent you all a clue:

Using the formulae
and

$$x = v \cos a \cdot t$$
$$y = v \sin a - \frac{1}{2} g t^2$$

as given last month, here is a trajectory demonstrator programme which could be adapted:

```
10 ::
20 input 'velocity & angle!' v,a
40 a=rad(a)
50 g=9.81
60 tmax=2*SIN(a)/g
65 CLS : SCALE v * tmax,0,0
70 for t=0 to tmax step .1
80 x=v*COS (a)*t
90 y=v*SIN(a)*t-.5*g*t*t
100 POINT x,y
110 end for t
120 ::
```

End of listing.

Start with $v = 75$ and $a = 50$ for the INPUT values at line 20 and experiment from there.

Remember that the closing date for entries is midnight on February 28th 2007. All entries to Sarah Gilpin, QUANTA Secretary. (see inside front cover for address etc.)

I recently became the proud owner of a Relisys TL565 LCD 15 inch monitor, courtesy of my (she claims) long-suffering wife. This monitor has an optimum resolution of 1024 x 768 pixels and was intended for use with my PC and QPC2, where it worked first time with no problems.

It has small stereo speakers built in with plenty of volume, although rather tinny in quality compared to most computer speakers. Picture quality is great, text is much sharper and easier on the eye than my old CRT monitor (less eye strain!) and of course takes much less space than the old CRT monitor. It's black and silver (there is a light grey or all black version available too). Above all, it cost less than 100 pounds from eBay.

I wondered if it could be used with my Minis-QL, which is based on an Aurora motherboard, with a Super Gold Card, SuperHermes Lite, MPlane and Qubide.

Well, the sound part was easy. A simple 3.5mm jack lead connected the Aurora's sound output to the monitor. The volume control allowed the beeps and keyboard clicks to be as loud as I needed.

Pictures weren't quite so easy. So here's my tale of how I got the Aurora to work with an LCD monitor, to save anyone else going through the same trauma.:

Problem 1: LCD monitors have one optimum display resolution. Anything else is scaled and looks poor or awful.

Problem 2: Most 15 inch LCD monitors are 1024x768 resolution, which the Aurora should be able to display in mode 4 (it doesn't have enough video memory to display 256 colours in 1024x768).

Problem 3: Most attempts to display anything other than 1024x768 ended with part of the Aurora display off-screen, rather like putting a QL in monitor windows mode with a TV set.

Problem 1 basically meant that the monitor preferred a 1024x768 display. This is 4:3 aspect ratio. Attempts to display 800x600 or 640x350 gave what can only be referred to as awful, unreadable pictures which either didn't fit the screen or were scaled in such a way that parts of letters went

Quanta Magazine

missing - imagine trying to read text where the vertical lines were missing in most letters. Resolutions which worked best were those which were multiples of 256 pixels across and 128 pixels down in a 4:3 aspect ratio. This meant that 512x384, 640x480 and 768x576 gave reasonable results, but 512x256, 640x320, 640x350 and even 800x600 gave very poor results.

Problem 2 was unexpected. At first, I couldn't get the Aurora into 1024x768 pixel mode. DISP_SIZE 1024,768 actually resulted in a 1024x576 display (confirmed with PRINT SCR_XLIM,SCR_YLIM). Ah well, if all else fails, read the manual, and find out that there's a set of three jumper links listed in Appendix A of the manual. These are near the centre of the motherboard, referred to as System Configuration Jumper Block in the manual. By default, these are set to the SVGA monitor, interlace disabled. This mode limits the display resolution to 1024x576 maximum. The logical step seemed to be to change this to Interlace Enabled setting, which only resulted in a blank screen. So I tried the Multisynch Monitor setting and this worked. I got a mode 4 1024x768 screen, but ran into problem 3.

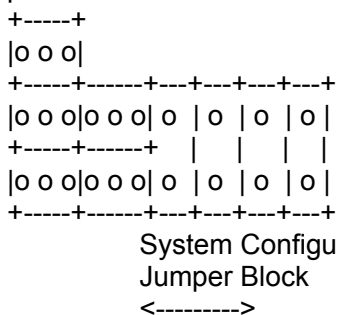


FIGURE 1 - Aurora System Configuration Jumper Block. The part that we are interested in is the set of three links on the right of the picture. To set Multisynch mode, join the rightmost pins with a jumper link and leave the two pairs of pins to the left of those unconnected.

Problem 3 was a strange one and the hardest to resolve. No matter what screen resolution the Aurora was set to, the monitor tried to auto adjust and parts of the display fell off screen. There seemed to be no logic to it - sometimes the display would be normal, sometimes not. I eventually realised that the auto-adjust seemed to set the visible display to the largest non-black section. If I set the Aurora to 640 x 480 only about 512 pixels across was shown. If I set a border around the entire screen, or set a background colour which wasn't black, then hit Auto Adjust on the monitor, normality was restored. So from this I concluded that Auto Adjust works by sensing non-black areas of the picture. So the easiest way around this is to set a white border around the entire screen.

Quanta Magazine

WINDOW #0,SCR_XLIM,SCR_YLIM,0,0 : BORDER #0,1,255 then hit Auto Adjust via the monitor's front panel buttons if it didn't set itself automatically.

Once the Aurora is set correctly, it's then just a matter of fooling the monitor into auto-adjusting correctly. Once you get it working, the 1024 x 768 Aurora display is fantastically clear on the monitor. And the MinisQL case is the right size to sit under the monitor, and the monitor is light enough to sit on the MinisQL case, so it all makes for a rather nice and compact system.



ROBOTICS ON THE QL (Part 2)

David Buckley

Last month I introduced my Zero2 Turtle and my QL program to control it. So how do we control a Turtle, in effect a robot vehicle? Well Zero2 was conceived in the days when embedded microcontrollers were not so easy to use as nowadays so Zero2 doesn't use one. The controlling computer (a QL, or any other PC with a serial port, eg a Spectrum) sends bytes at 4800 baud which are received by Zero2 with an onboard 6402 UART, the parallel output of which is routed through some inverters, latches and darlington drivers to control the stepper motor for each wheel and the stepper motor for the pen, as well as the lights and horn.

Quanta Magazine

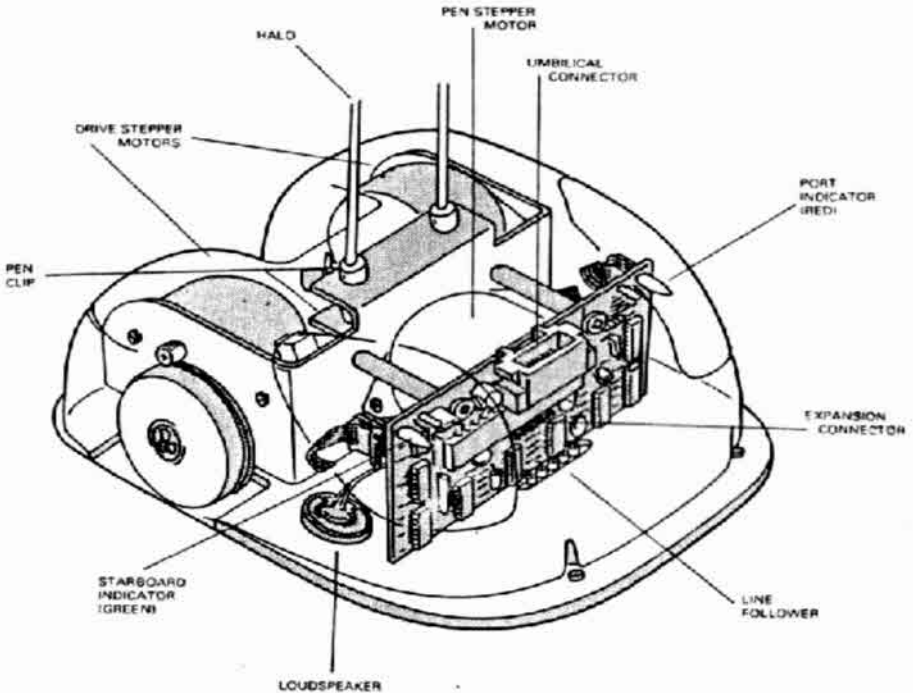


Figure 1

Figure 1 shows the main components of Zero2 and shows the items mentioned above.

Zero 2 is very easy to program, the serial port (either RS232 or RS423) must be to 4800 baud one stop bit and no parity.

Every function of Zero2 can be controlled and monitored by the computer. To communicate directly, it must be told which device is to be involved, and what that device is to do. Each device has a numeric *address*, to which the control *data* is sent. Valid addresses are from 0 to 7, though not all addresses have anything there.

Address	Effector device	Sensor device
0	Drive motors	Line/Edge detectors
1	Pen motor	Left bumper
2	Indicators	Right bumper
3	Speech	
4	Speech/Sound	
5	Sound	

Quanta Magazine

Once you have identified the address of the device you wish to control, you need to send this number together with the data to the robot. This is sent as one value, The address is in the high nibble and the data in the low nibble, ie multiplying the address by 16 and add the data.

Data values can be between 0 and 15. For example, to send a data value of 5 to the indicators:

$$\begin{aligned} \text{Address} &= 2 \times 16 &= 32 \\ \text{Data} &= 5 &+ \\ \text{Value sent to port} &= 37 \end{aligned}$$

The data to be sent varies for each device, as can be seen by a quick look at the next section. This is because each of the controls in the robot is connected to one of four electronic switches, each of which can be on or off. The data you send decides which switches are on and which are off. The following table shows which switches are on for each of the data values (1 means on, 0 means off):

<i>Data</i>	<i>Switch</i>	<i>Data</i>	<i>Switch</i>
	3210		3210
0	0000	8	1000
1	0001	9	1001
2	0010	10	1010
3	0011	11	1011
4	0100	12	1100
5	0101	13	1101
6	0110	14	1110
7	0111	15	1111

i.e. simple binary encoding.

Each motor is controlled by two switches, switches 2 and 3 for the left motor, switches 0 and 1 for the right motor. To make each motor move forwards you need to send instructions in the order 01,00,10,11. By checking the above table, you can see that to move the right motor only, you can send the data as 1,0,2,3. To drive the left motor only, you send 4,0,8,12. To drive both motors forward send 5,0,10,15. To drive the motors backwards send the data in the reverse order. To turn the motors in opposite directions send 6,0,9,15 or 9,0,6,15. This is a lot of numbers, but on closer examination they are all obtained from the original 1,0,2,3 and 4,0,8,12 by adding the value for the step position of one motor to that for the other. You start at one end for one direction, and the opposite end for the other.

Quanta Magazine

$$\begin{array}{r} 1\ 0\ 2\ 3 \\ +\ 4\ 0\ 8\ 12 \\ \hline 5\ 0\ 10\ 15 \end{array} \quad \begin{array}{l} (3) \\ (12) \end{array} \quad \begin{array}{r} 2\ 0\ 1\ 3 \\ \hline 4\ 0\ 8\ 12 \\ 6\ 0\ 9\ 15 \end{array}$$

Each time you send a data code to address 0, one or both motors will move one step. If the data is out of sequence the motors will not operate properly and will probably just judder. Each step moves the wheel by 0.5mm.

Alternatively to move 4 steps at a time, then by always ending on 15 the codes will remain in sequence. I.e.

forward - 5, 0, 10, 15,
backward - 0, 0, 5, 15,
right - 6, 0, 9, 15,
left - 9, 0, 6, 15.

The pen is controlled by the same kind of motor as the wheels, and the same codes are used to control it, except that only switches 0 and 1 are involved. One step of the motor is 1/48 of a complete turn. Normally you will want to move the pen motor half way round each time which is 24 steps, so you would send the full sequence of steps six times. Remember that the pen is at address 1, ie the high nibble of the data=1. The position of the pen should be initialised to 'up' by sending individual steps until the pen is 'up', then the software and the pen will be synchronised. There is no need to send the codes to reverse the motor. 24 forward steps from 'up' will lower the pen to 'down' and another 24 forward steps will raise it to 'up' again.

The LEDs and horn are connected to the switches at address 2 as follows;

Green LED	Switch 0
Red LED	Switch 1
Low Horn	Switch 2
High Horn	Switch 3

To switch on the red LED, for instance, you need to send the 'on' code for switch 1. Unfortunately, if you send just that, you will switch off the other LED and horn if they were on. Every time you send a value to an address it replaces the previous value. You must keep a note of what was previously sent to the address and make allowances when sending fresh data. Assume the low horn and green LED are already on, and you want to turn on the red LED. The last data sent to address 2 was 5 (0101 - see

Quanta Magazine

table). The code for red LED on is 2 (0010), Add these numbers to get 7 (0111) which will give you the new data. Add on the address x 16, send it to the robot and on comes the red LED. Now, suppose you want to switch off the green LED. The code for green LED on is 1. Subtract this from the last data value of 7 to get the new data.

Each time a value is sent to a robot address. Zero 2 will send back a code relating to any sensors connected to that address. Address 0 can have up to six switches connected to it but the other addresses can only have up to four each.

Only address 0 has sensors connected, these being the three line follower sensors, connected to switches 0 to 2 (switch 3 is always off).

Whenever either drive motor is stopped, the line follower sensors will send back a value from 0 to 7 which can be used to allow Zero 2 to follow a line. A value of 0 means that all the sensors see a dark surface, while 7 means they all see a bright one.

Most BASICs of the time were fairly slow when dealing with the real world, so rapid response were never a problem when programming Zero 2 directly from BASIC. When running the Zero2 Basic Control-program under QPC2 under XP some strange things seem to happen with the Serial port and its buffer and Zero2 runs slow and judders. Quitting and rerunning the program clears the problem as does quitting and restarting QPC2. This never happened on a QL. A very simple program to drive the robot forward 100 steps is given below, assuming you have already set the baud rate:

```
100 zchan=3      :REMark for serial port to robot seri1
110 baud_rate=4800
120 OPEN #zchan,seri:BAUD baud_rate
130 DIM mtrcmd(4)
140 mtrcmd(1) =5: mtrcmd(2) =0: mtrcmd(3) =10: mtrcmd(4) =15
150 FOR dist=1 TO 100/2 :REMark because 4 steps are 2mm
160   FOR step=1 TO 4
170     PRINT #zchan, CHR$( mtrcmd(step) );
180   NEXT step
190 NEXT dist
200 END
```

Note the ';' at the end of the print statement to prevent a LF being sent. LF=10 which will upset the sequence of codes sent to the motors.

Quanta Magazine

When writing to Zero 2 each device, or device group, has an address which forms the first three bits of the upper nibble. These are allocated as follows.

Address	Device	Data bit allocation			
0	Drive Motors	D3 left (port) motor	D2	D1 right (starboard) motor	D0
1	Pen	D3 for future allocation	D2	D1 Pen lift motor	D0
2	Indicators	D3 Horn High Tone	D2 Horn Low Tone	D1 Left LED	D0 Right LED
3-7		for future use		for future allocation	

In a small Stepper motor the control sequence codes are normally:

Binary	Decimal
0101	5
1001	9
1010	10
0110	6
0101	5

In Zero2 this is generated from two data lines with two inverters

**		I*I*	(I=inverted signals)
1 1	3	0101	5
0 1	1	1001	9
0 0	0	giving 1010	10
1 0	2	0110	6
1 1	3	0101	5

Each full motor step results in the robot travelling 0.5mm

[Once again space has run out and we will have to postpone the conclusion of this article until the next issue Ed]

Quanta Magazine

[There are no changes to any of the Sub Groups - Please see Issue 5 for Sub Group details Ed]

[Due to the lateness of this Issue, the QUANTA News feature has been omitted. Anything which can still be classified as NEWS will be included next Issue Ed]

Quanta Magazine

Feeling out on a limb?
Reach out for QBranch.
Suppliers of computer solutions
Hardware and Software



Just a small selection from our catalogue

QDT - The QL Desktop	£ 41.50 + P&P
New version available now. 'Drag and Drop' & improved icon editor. File Manager coming during the summer with many new features.	
EASYPTR parts 1&2 version 4 - high colour	£ 41.00 on CD with manual in Adobe format
EASYPTR parts 1&2 upgrade	£ 27.50 on CD (send master disk)
QPCPrint - print to any printer from QPC2 (needs v3.3x QPC2)	£ 27.50 on CD with manual in Adobe format
QPC2 v 3.32 (SMSQ/E v3.12)	£ 42.00
Upgrade from version 3.xx	£ Free. Send Master disk and return P&P
Upgrade from v2.xx	£ 13.90
Upgrade from v 1.xx	£ 34.00
SMSQ/E v3.12 for all systems out now. New features include background updating. Aurora High Colour version is now included as standard.	
Keyboard mebranes few left	£ 17.50
New Super/Gold Card Batteries	£ 10.00 + P&P
Aurora Motherboards - new	£ 65.00

20 Locks Hill, Portslade, Sussex. BN41 2LB. UK
Tel: +44 (0)1273 386030 Fax: +44 (0)1273 430501
email: sales@qbranch.demon.co.uk skype : royqbranch
web : <http://www.qbranch.demon.co.uk>

TF Services

Compswitch

A UK 4-way trailing socket designed to switch off computer peripherals automatically when the computer is switched off, or (in the case of an ATX computer) when it auto-powers down. *Compswitch* has one control socket, and three switched sockets. Can be used with lights/fufl/monitors—ie a QL monitor can be used as a switch control.

Cost **£24**

superHermes

A major hardware upgrade for the QL

All Hermes features (working ser1/2 at 19200, independent baud rates/de-bounced keyboard/keyclick) IBM AT kbd I/F // HIGH SPEED RS232 at 57600// serial mouse port and 2 other RS232 inputs// 3 I/O lines // EEPROM

Cost (including manual/software) **£90** (€92/€93)
 IBM AT UK layout Keyboard..... **£11** (€13/€15)
 Serial mouse..... **£8** (€8.50/€9)
 Capslock/scrolllock LED **£1** (€1.50/€1.50)
 Keyboard or mouse lead..... **£3** (€3.50/€3.50)
 High speed serial (ser3) lead..... **£4** (€4.50/€4.50)

Hermes available for **£25** (€26/€27) Working ser1/2 and independent input, debounced keyboard.

SuperHermes LITE: All Hermes features (see above) + an IBM AT keyboard interface only.

Cost (incl keyboard lead) **£53** (€54/€55)

QL REPAIRS (UK only)

Fixed price for unmodified QLs, excl microdrives. QLs tested with Thom-EMI rig and ROM software.

£27 incl 6 month guarantee

Minerva

The ORIGINAL system operating system upgrade

OTHER FEATURES COMMON TO ALL VERSIONS

DEBUGGED operating systems/ autoboot on reset of power failure/ Multiple Basic/ faster scheduler- graphics (within 10% of lightning) - string handling/ WHEN ERROR/ 2nd screen/ TRACE/ non-English keyboard drivers/ "warm" fast reset. V1.97 with split OUTPUT baud rates (+Hermes) & built in Multibasic.

First upgrade free. Otherwise send **£3 (+£5 for manual if reqd)**. Send disk plus SAE or two IRCs

MK1...£40 (€41/€43) MKII...£65 (€66/€67)

MINERVA RTC (MKII) + battery for 256 bytes ram. CRASHPROOF clock & FC bus for interfacing. Can autoboot from battery backed ram. Quick start-up.

QL RomDisq

Up to 8 mbyte of flash memory for the QL

A small plug in circuit for the QL's ROM port (or Aurora) giving 2, 4 or 8 mbytes of permanent storage- it can be thought of as a portable hard disk on a card, and reads at some 2 mbytes per second.

Think of it - you could fully boot an expanded QL, including all drivers/SMSQ etc off RomDisq at hard disk speed with only a memory expansion needed.

2 mbytes RomDisq..... **£39** (€40/€41)
 4mbytes RomDisq..... **£65** (€66/€67)
 8 mbytes RomDisq..... **£98** (€99/€100)
 Aurora adaptor..... **£3** (€3.50/€4)

MPLANE

A low profile powered backplane with ROM port

A three expansion backplane with ROM port included for RomDisq etc. Aurora can be fitted in notebook case and powered off single 5V rail - contact QBranch for details. Two boards (eg Aurora and Gold Card/Super Gold Card/Goldfire fixed to base. Suitable for Aurora (ROM accessible from outside) & QL motherboard in tower case. Specify ROM facing IN towards boards, or OUT towards back of case.

Cost **£34** (€35/€36)

I2C INTERFACES

Connects to Minerva MKII and any Philips I²C bus

Power Driver Interface 16 I/O lines with 12 of these used to control 8 current carrying outputs (source and sink capable)

2 amp (for 8 relays, small motors)..... **£40** (€43/€44)

4 amp total (for motors etc)..... **£45** (€48/€50)

Relays (8 3a 12v 2-way mains relays (needs 2a power driver)..... **£25** (€28/€29)

Parallel Interface Gives 16 input/output lines. Can be used wherever logic signals are required..... **£25** (€27/€28)

Analogue Interface Gives eight 8 bit analogue to digital inputs (ADC) and two 8 bit digital to analogue output (DAC). Used for temp measurements, sound sampling (to 5 KHz), x/y plotting..... **£30** (€31/€32)

Temp probe (-40°C to +125°C)..... **£10** (€10.50/€11)

Connector for four temp probes..... **£10** (€10.50/€11)

Data sheets..... **£2** (€2.50/€3)

Control software & manual (for all I/F)..... **£2** (€2.50/€3)

QL SPARES

Keyboard membrane..... **no longer on sale**
 1377 PAL..... **£37** (€3.50/€4)
 Circuit diagrams..... **£3** (€3.50/€4)
 68008 cpu or 8049 IPC..... **£8** (€8.50/€9)
 8301/8302 or JM ROM or serial lead..... **£10** (€10.50/€11)
 Power supply (see mail overseas)..... **£12** (€17/€21)
 Other components (sockets etc) also available

Prices include postage and packing (Airmail where applicable) Prices are: UK (Europe/Rest of world) Payment by cheque drawn on bank with UK address, postal order or CASH! I can no longer accept card payments as UK only does TDQ transaction. SAE or IRC for full list and details 31 Oct 02

29 Longfield Road, TRING, Herts, HP23 4DG
 Tel: 01442-828254 Fax/BBS: 01442-828255
 tony@firshman.co.uk http://www.firshman.co.uk